Research study of various Deep Learning models:

1. CNN
2. ResNet
3. Yolo (different versions)
   Compare and contrast the pros and cons of each of the models and make an analysis report about the complete working and in-depth architecture. List down the features of each model which according to you are useful.

1. Convolutional Neural Network (CNN):

Working and in-depth architecture:

Working:
- Input: A CNN takes an input image or a batch of images.
- Convolutional Layers: The first layer in a CNN is usually a convolutional layer. It applies a set of filters or kernels to the input image, performing convolution operations. Each filter detects specific features, such as edges or textures, and generates a feature map by sliding across the image.
- Activation Function: Non-linear activation functions, like ReLU (Rectified Linear Unit), are applied to the feature maps to introduce non-linearity, allowing the network to learn complex patterns.
- Pooling Layers: After the activation function, pooling layers are typically used to reduce the spatial dimensions of the feature maps while retaining important information. Common pooling operations include max pooling or average pooling.
- Fully Connected Layers: The output of the pooling layers is flattened into a vector and fed into fully connected layers. These layers are similar to those in traditional neural networks and are responsible for making predictions based on the learned features.
- Output: The final layer of the CNN produces the predicted output, which could be class probabilities (in classification tasks) or bounding boxes and class probabilities (in object detection tasks).

Architecture:

1. Convolutional Layer

This layer is the first layer that is used to extract the various features from the input images. In this layer, the mathematical operation of convolution is performed between the input image and a filter of a particular size MxM. By sliding the filter over the input image, the dot product is taken between the filter and the parts of the input image with respect to the size of the filter (MxM).

The output is termed as the Feature map which gives us information about the image such as the corners and edges. Later, this feature map is fed to other layers to learn several other features of the input image.

2. Pooling Layer

In most cases, a Convolutional Layer is followed by a Pooling Layer. The primary aim of this layer is to decrease the size of the convolved feature map to reduce the computational costs. This is performed by decreasing the connections between layers and independently operates on each feature map. Depending upon method used, there are several types of Pooling operations. It basically summarises the features generated by a convolution layer.

In Max Pooling, the largest element is taken from feature map. Average Pooling calculates the average of the elements in a predefined sized Image section. The total sum of the elements in the predefined section is computed in Sum Pooling. The Pooling Layer usually serves as a bridge between the Convolutional Layer and the FC Layer.

3. Fully Connected Layer

The Fully Connected (FC) layer consists of the weights and biases along with the neurons and is used to connect the neurons between two different layers. These layers are usually placed before the output layer and form the last few layers of a CNN Architecture. In this, the input image from the previous layers are flattened and fed to the FC layer. The flattened vector then undergoes few more FC layers where the mathematical functions operations usually take place. In this stage, the classification process begins to take place. The reason two layers are connected is that two fully connected layers will perform better than a single connected layer.

4. Dropout

Usually, when all the features are connected to the FC layer, it can cause overfitting in the training dataset. Overfitting occurs when a particular model works so well on the training data causing a negative impact in the model's performance when used on a new data.

To overcome this problem, a dropout layer is utilised wherein a few neurons are dropped from the neural network during training process resulting in reduced size of the model. On passing a dropout of 0.3, 30% of the nodes are dropped out randomly from the neural network.

5. Activation Functions

Finally, one of the most important parameters of the CNN model is the activation function. They are used to learn and approximate any kind of continuous and complex relationship between variables of the network. In simple words, it decides which information of the model should fire in the forward direction and which ones should not at the end of the network. There are several commonly used activation functions such as the ReLU, Softmax, tanH and the Sigmoid functions.

Pros:

- CNNs are quite good at recognising images and videos.
- They learn hierarchical representations automatically from data.
- Through convolutional and pooling layers, CNNs may detect spatial and temporal connections.
- Because of their translation invariance property, they can accommodate inputs of varying sizes.
- End-to-end CNN training eliminates the need for manual feature engineering.

Cons:

- CNNs can be expensive to compute, especially for huge datasets.
- To generalize well, they need a lot of training data.
- Fine-grained details and collecting long-range dependencies may be difficult for CNNs to handle.

Key Features:

- Convolutional layers: By applying filters to the input data, these layers help the network identify regional trends.
- Pooling layers: These layers reduce the spatial dimensionality by downsampling the feature maps.
- Activation functions: ReLU (Rectified Linear Unit) and sigmoid are two frequently used activation functions in CNNs.
- Layers that are fully connected: These layers link every neuron from one layer to the next, allowing for complex reasoning.

2. Residual Network (ResNet):

Working and in-depth architecture:

Working:

- Input: The input to ResNet is typically an image or a batch of images.
- Convolutional Layers: ResNet starts with a few convolutional layers to extract features from the input image.
- Residual Blocks: The key component of ResNet is the residual block. Each block consists of multiple convolutional layers followed by an element-wise addition operation. The addition is performed between the input to the block (identity) and the output of the convolutional layers. This forms a "shortcut" or "skip connection," allowing the gradient to flow directly through the network.
- Identity Mapping: The skip connection allows the network to learn the residual mapping, i.e., the difference between the input and output of the convolutional layers. The network can then learn to refine the features rather than starting from scratch in each layer.

- Deep Stacking: ResNet encourages the stacking of residual blocks, enabling the network to be significantly deeper while still maintaining gradient flow and avoiding the degradation problem.
- Global Average Pooling: After several residual blocks, a global average pooling layer is applied to reduce the spatial dimensions of the feature maps to a vector representation.
- Fully Connected Layer: A fully connected layer takes the vector representation and produces the final predictions.
- Output: The final layer provides the predicted output, which could be class probabilities in classification tasks or bounding boxes and class probabilities in object detection tasks.

Architecture:
ResNet architectures are often referred to by the number of layers or blocks they contain, such as ResNet-18, ResNet-34, ResNet-50, etc. Deeper versions like ResNet-101 and ResNet-152 also exist. These architectures differ in the number of residual blocks and the number of convolutional layers within each block. Pre-trained versions of ResNet are widely used for various computer vision tasks and are available for transfer learning purposes.

Pros:
-ResNet solves the vanishing gradient issue, enabling deeper networks. It also adds residual connections, which help gradients flow through skip connections.
- As the network's depth rises, ResNet can pick up more abstract elements.
- It performs at the cutting edge in terms of object detection, image classification, and other computer vision tasks.

Cons:
- ResNet designs may be more complicated and challenging to train than conventional CNNs.
- Increasing model depth could result in higher memory and processing expenses.

Key Features:
Residual blocks are used to speed up the training of deep networks since they contain skip connections that let gradients skip over specific layers.
- Shortcut connections: These connections help the gradient flow and let the model learn identity mappings.
- Bottleneck architecture: Bottleneck blocks, which use 1x1 convolutions to lower the computational cost, are frequently used in ResNet topologies.

3. You Only Look Once (YOLO):

Working and in-depth architecture:

Working:

- Image Division: The input image is divided into a grid, usually of fixed size, forming cells. Each cell is responsible for predicting the bounding boxes and class probabilities for the objects present in its region.
- Anchor Boxes: YOLO uses anchor boxes of different sizes and aspect ratios, which are predefined bounding box priors. These anchor boxes are used to predict the bounding box coordinates relative to each cell.
- Convolutional Layers: YOLO typically consists of multiple convolutional layers to extract features from the input ima YOLO may struggle with accurately localizing small objects in an image.   It can produce lower accuracy compared to some region-based approaches. YOLO relies on predefined anchor boxes, which may limit its ability to handle objects of various sizes and aspect ratios.ge.
- Predictions: Each cell predicts multiple bounding boxes (usually 2 or more) along with class probabilities. The predictions include the coordinates of the bounding boxes (x, y, width, height) and the confidence score, indicating the presence of an object.
- Non-Maximum Suppression (NMS): To eliminate duplicate detections and improve precision, a post-processing step called NMS is applied. It selects the most confident bounding boxes and suppresses overlapping boxes with lower confidence scores.
- Output: The final output consists of the selected bounding boxes and their associated class labels.

Architecture:
The YOLO architecture typically consists of a backbone network, which is usually a pre-trained CNN (such as DarkNet or ResNet), followed by several convolutional layers. The final layers include convolutional layers with larger receptive fields to capture global context and make predictions. YOLO has undergone several iterations, with YOLOv3 and YOLOv4 being widely used versions.

Pros:

- YOLO is renowned for its ability to detect objects in real-time.
- It is quicker than some other systems since it completes object detection and categorization in a single pass.
- Multiple object detection is supported by YOLO, which also offers bounding box coordinates and class probabilities.
- It is extensively used for several purposes, such as robotics, autonomous driving, and video surveillance.

Cons:

-YOLO may struggle with accurately localizing small objects in an image.
-It can produce lower accuracy compared to some region-based approaches.
-YOLO relies on predefined anchor boxes, which may limit its ability to handle objects of various sizes and aspect ratios.

Key Features:

- Grid cell approach: YOLO predicts bounding boxes and class probabilities within each cell by dividing the input image into a grid of cells.
- Single pass architecture: YOLO efficiently accomplishes detection and classification in a single pass.
- Anchor boxes: To handle object localisation and classification, YOLO employs anchor boxes with predetermined sizes.

Other models

1. Gated Recurrent Unit (GRU): GRU is another variant of RNN that aims to address the vanishing gradient problem. It simplifies the architecture compared to LSTM while still capturing important information. GRUs have been used in various tasks such as language modeling, sentiment analysis, and speech synthesis.
2. Generative Adversarial Networks (GAN): GANs consist of two neural networks, a generator and a discriminator, which are trained in an adversarial manner. GANs are primarily used for generating new data instances, such as realistic images, text, and even videos.
3. Transformer: Transformers are attention-based models that have gained significant attention in natural language processing tasks. They excel in capturing long-range dependencies and have been utilized in machine translation, language understanding, and text summarization.
4. Deep Reinforcement Learning: Deep RL combines deep learning with reinforcement learning algorithms to train agents that can make sequential decisions. It has been applied to various domains, including game playing, robotics, and autonomous driving.
5. ResNeXt: The ResNeXt model incorporates the ResNet strategy of repeating layers but introduces an extensible, simple way to implement the split, transform, and merge strategy.
6. DenseNet is another popular ResNet variation, which attempts to resolve the issue of vanishing gradients by creating more connections.